

Execution of UML models

Present and Future of Research and Practice

Federico Ciccozzi, Ivano Malavolta, Bran Selic

Mälardalen University, Vrije University, Malina Software Corp.

Ericsson Modeling Days 2016





UML1(.x)

- In industrial practice, UML models have been used primarily for problem understanding(i.e., analysis) and documentation
- Nevertheless, several tools supporting executable variants of UML
- ***These tools historically relied on custom semantics (profiles)*** in combination with traditional third-generation programming languages (e.g., C++ or Java) for specifying detailed action code



Thereby..

- Tools not fully compliant with the UML standard
- (Industrial) end users forced into a potentially dangerous “**vendor lock-in**” predicament



The (r)evolution of UML2

- Semantically more precise than UML1 when it comes to execution aspects
- Provided with fUML, a formal specification of the executable semantics for a subset of UML2
- Provided with ALF, a textual action language based on fUML for compact and complete specification of complex behaviors including their algorithmic parts



So that..

*UML models based on UML2, fUML and ALF are fully executable provided that corresponding **methods and tools are available***



About this study: Why?

***Identify, classify and evaluate** the states of the art
and practice of solutions for executing models based
on the UML family of languages*



About this study: What?

- a classification framework for classifying, comparing, and evaluating solutions for UML models execution
- an up-to-date systematic review of the states of the art and practice in UML models execution
- an exploration of emerging research challenges and implications for future research and practice of UML models execution



About this study: How?

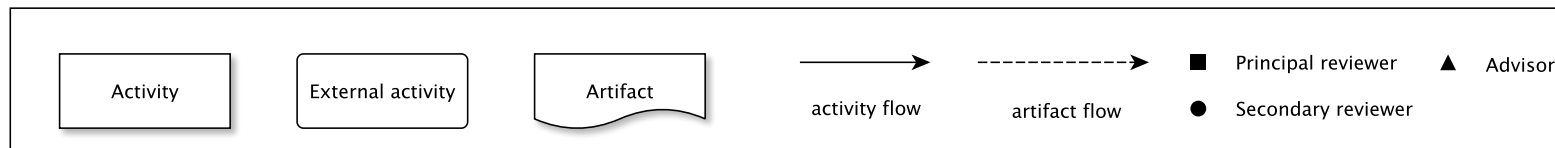
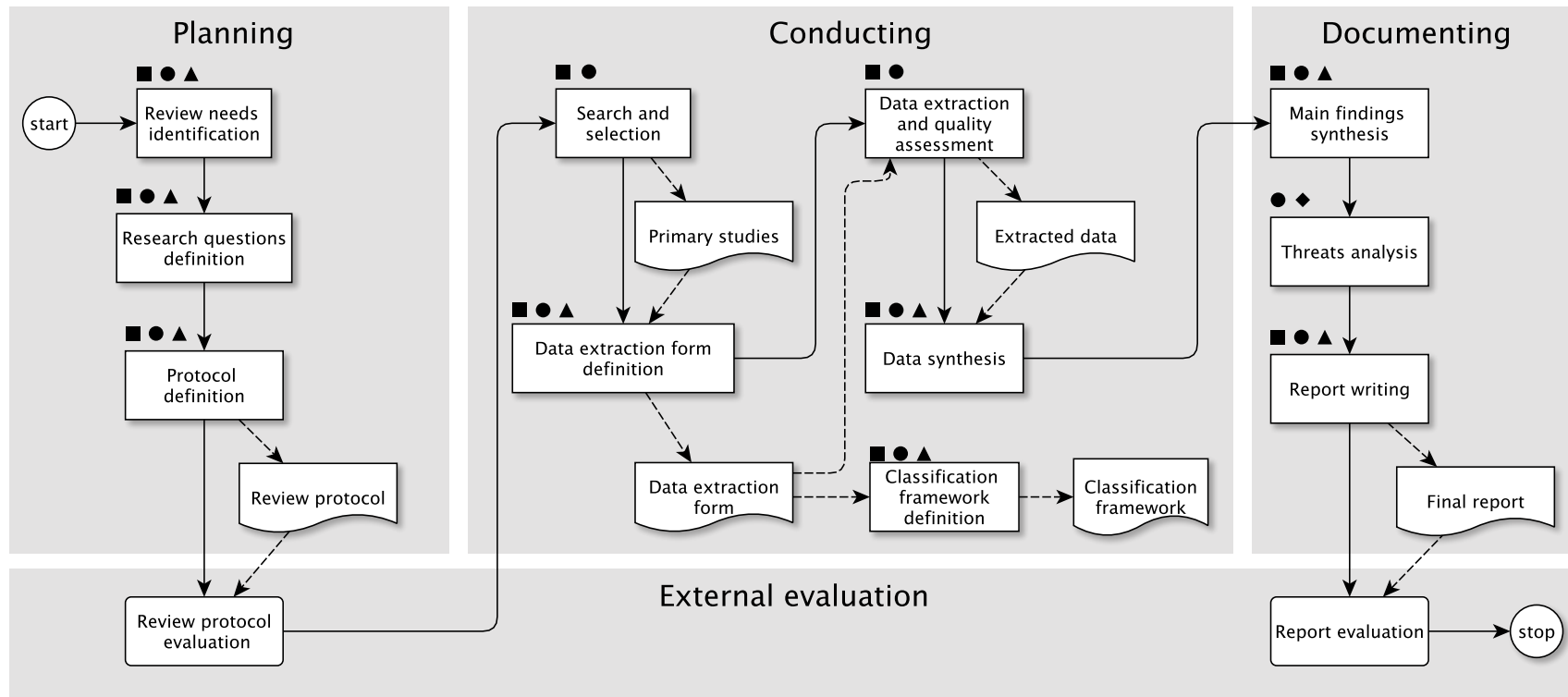
- Systematic review (SR)
- The first systematic investigation of the states of the art and practice of the execution of UML models
- Over 4,500 study candidates were scrutinized
- 70 studies were selected for answering the research questions that we identified
- Study report currently under revision at the IEEE Transactions on Software Engineering (TSE)



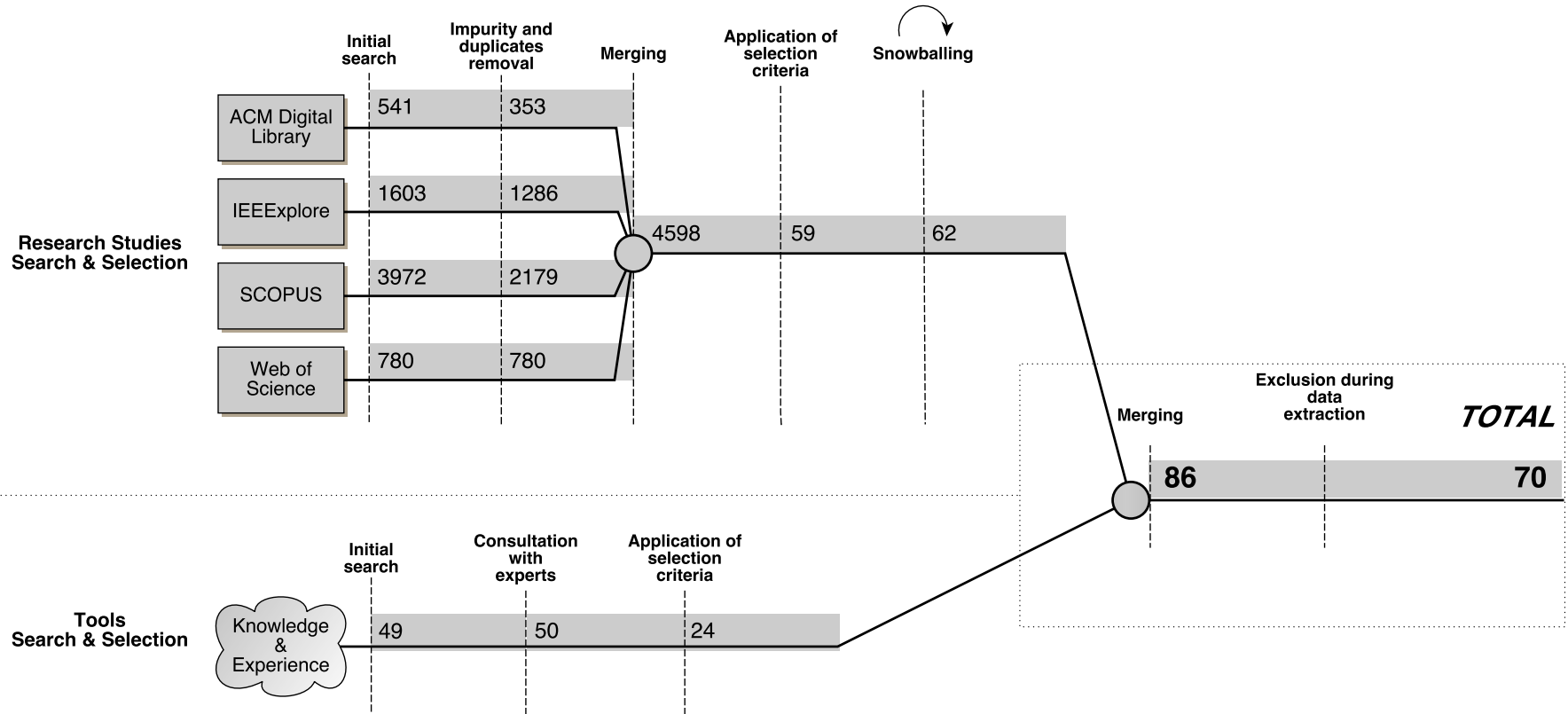
About this study: Who?

- **Principal researcher:** *Federico Ciccozzi*, involved in all the activities, i.e. planning the study, conducting it, and reporting
- **Research methodologist:** *Ivano Malavolta*, involved in (i) the planning phase of the study, and (ii) supporting the principal researcher
- **Advisor:** *Bran Selic*, provided guidance on key decisions and solving conflicts. He also supported the other researchers during data and findings synthesis activities

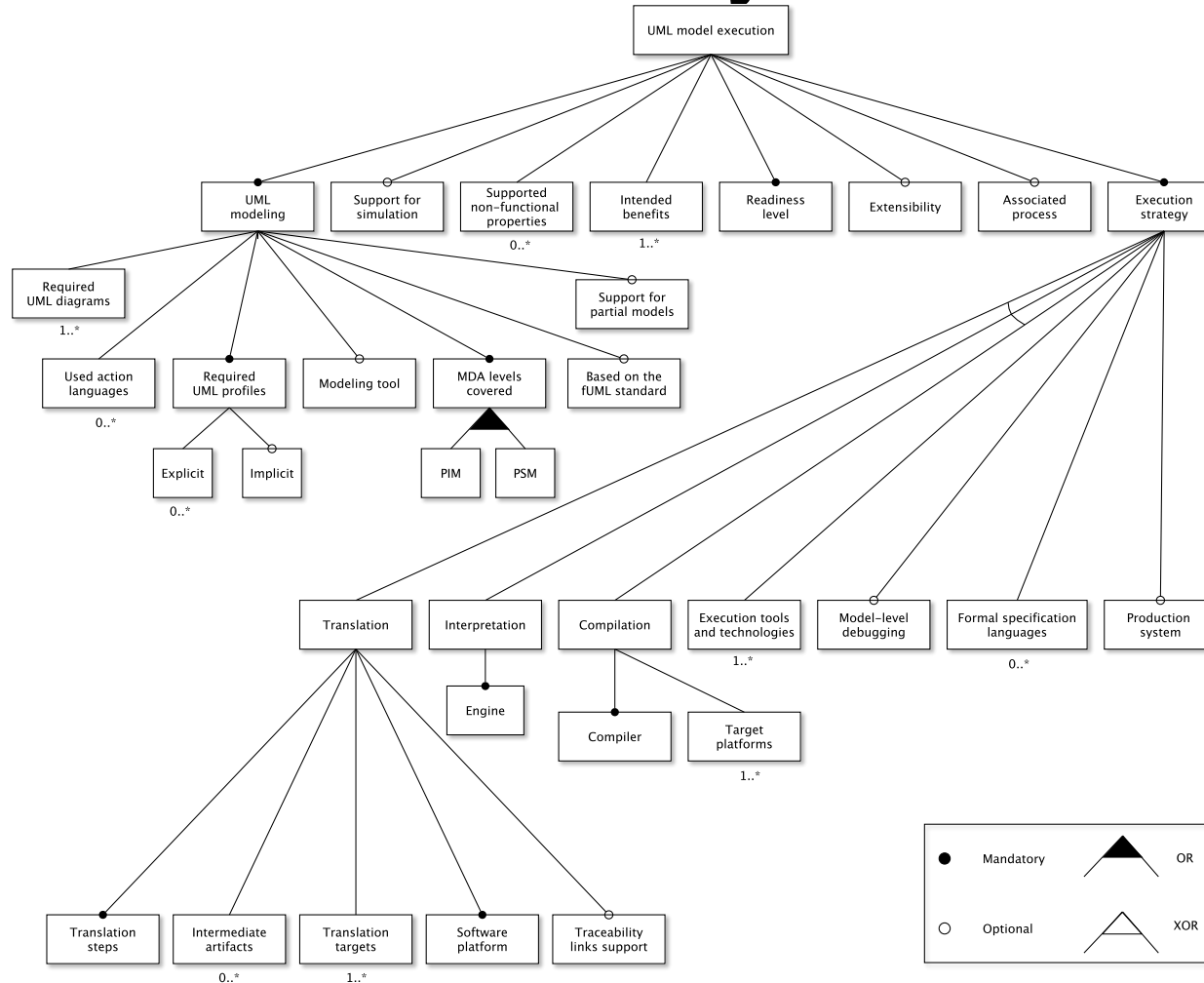
Our Systematic Review Process



Search and Selection



Data extraction and synthesis



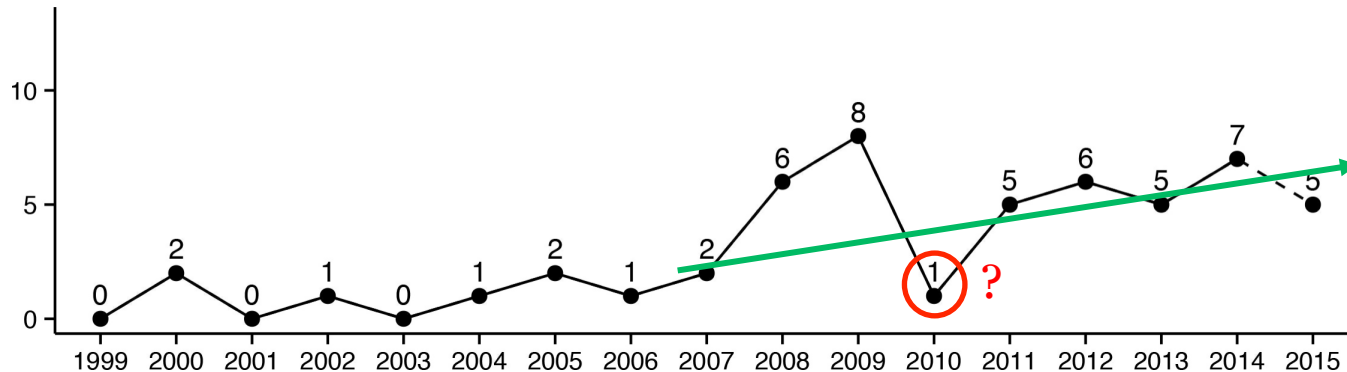


Results: Publication trends

- RQ1 – What are the publication trends of research studies pertaining to solutions for execution of UML models?

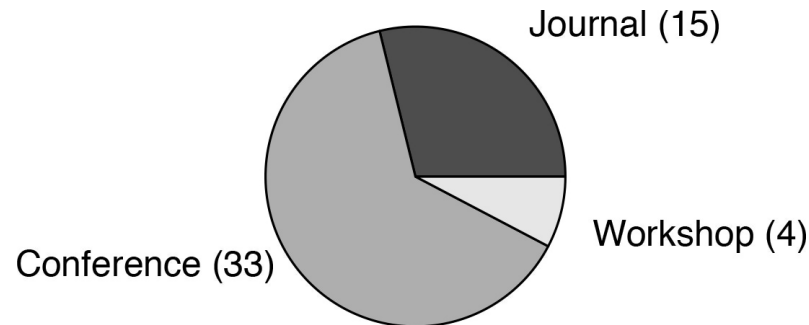
Results: Publication trends

- From 2008 there has been a growing scientific interest on UML models execution; this positive trend has been steady in the past 5 years (except 2010)



Results: Publication trends

- Conferences and journals are the most targeted publication venues, testifying that UML models execution is becoming a significant research theme





Results: Publication trends

- Research on UML models execution is spread across a large number of heterogeneous venues
- Researchers focus more on benefits and effects of models execution, rather than on the specific execution techniques

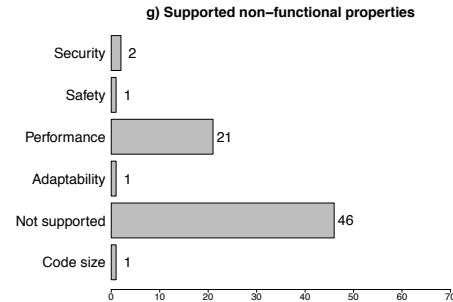
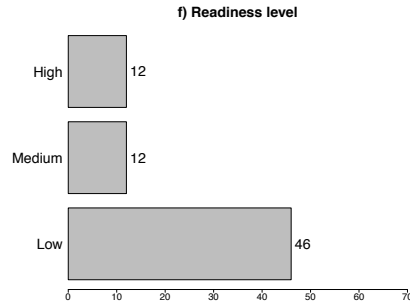
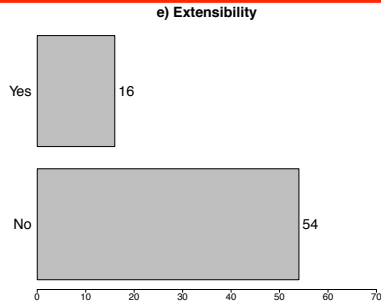
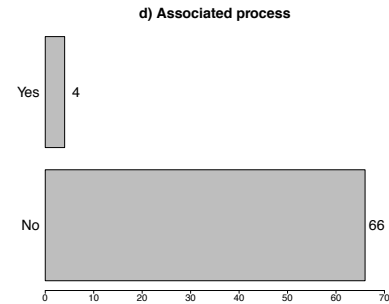
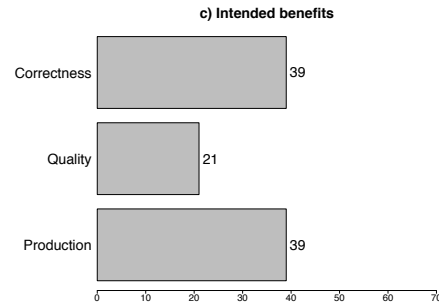
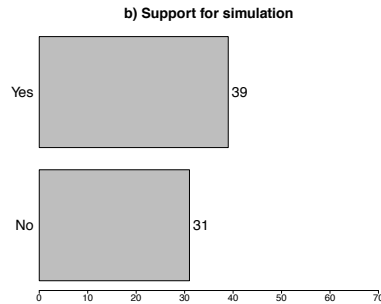
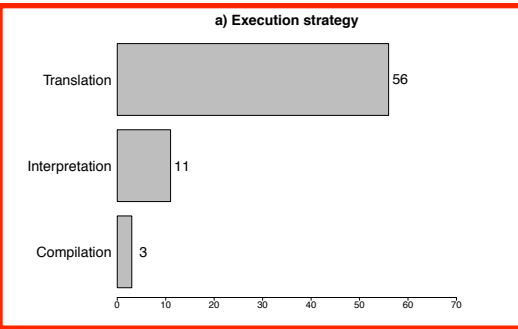


Results: Technical characteristics

- RQ2 – What are the technical characteristics of existing solutions for execution of UML models?

Results: Technical characteristics

- Solutions providing translational execution (i.e., code generation) outnumber interpretive and compilative solutions





Results: Technical characteristics

- Interpretive solutions are mainly addressing higher-level execution (e.g., for simulation)
- Compilative solutions leverage very limited subsets of UML



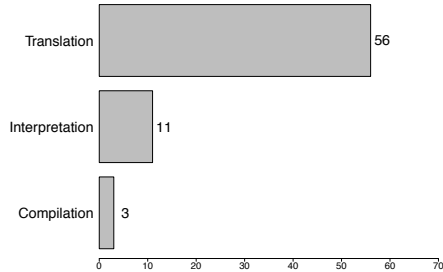
Results: Technical characteristics

- There is no solution for execution of partial UML models

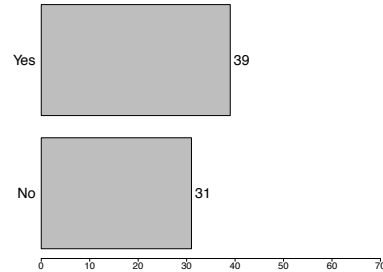
Results: Technical characteristics

- A very small amount of solutions explicitly provides mechanisms which enable extension and customization

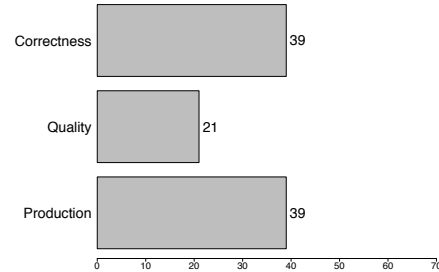
a) Execution strategy



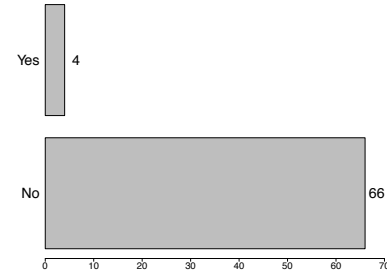
b) Support for simulation



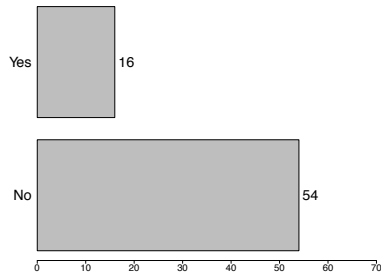
c) Intended benefits



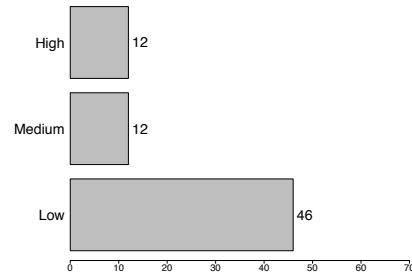
d) Associated process



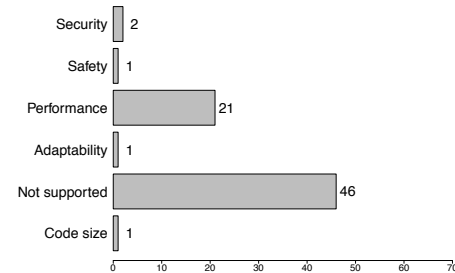
e) Extensibility



f) Readiness level

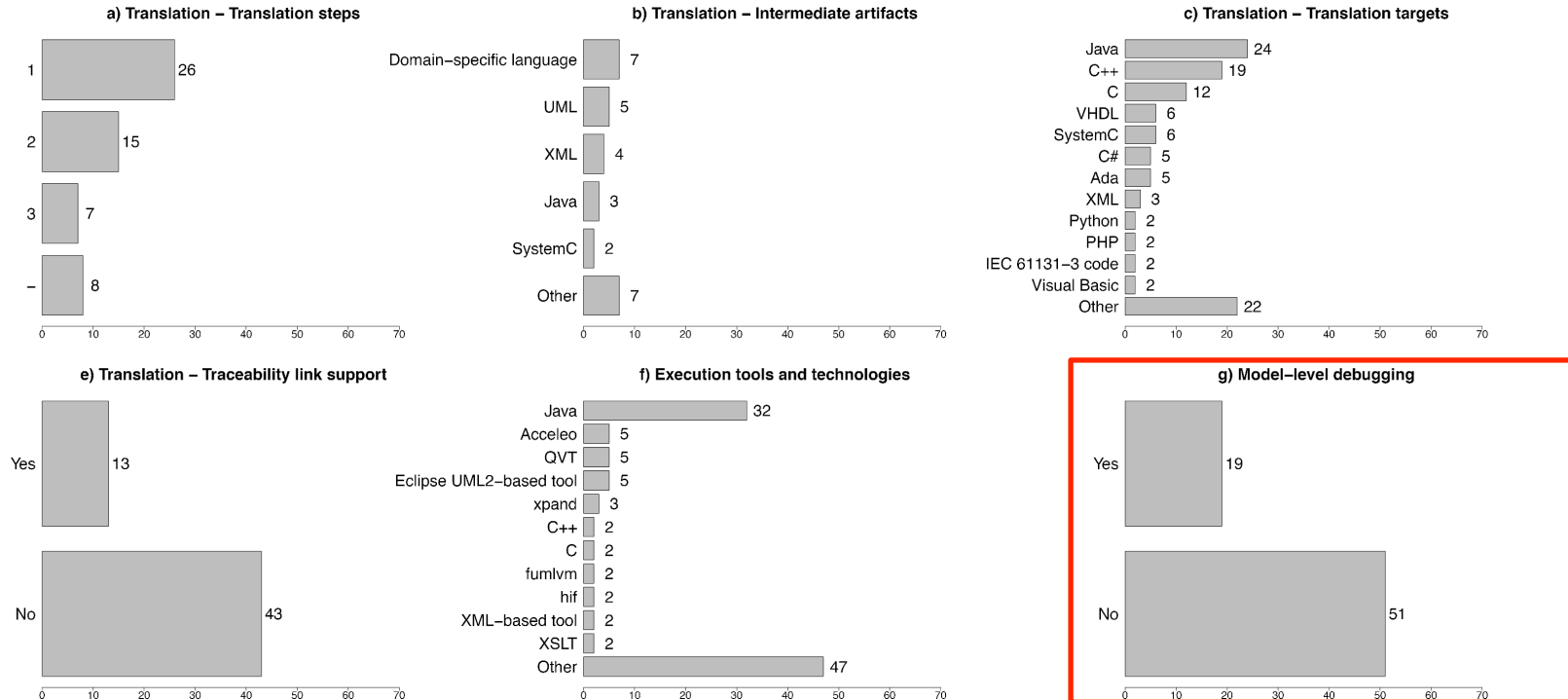


g) Supported non-functional properties



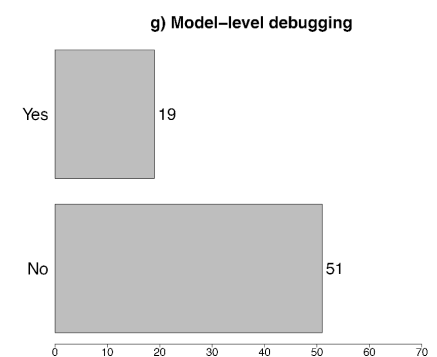
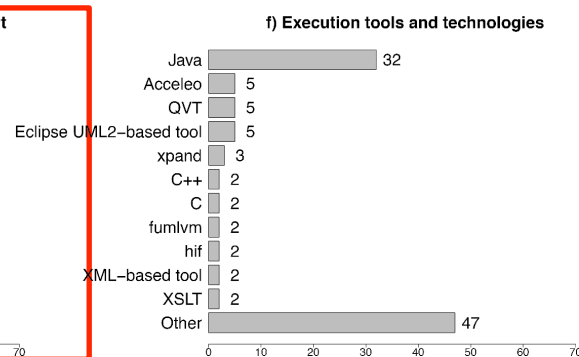
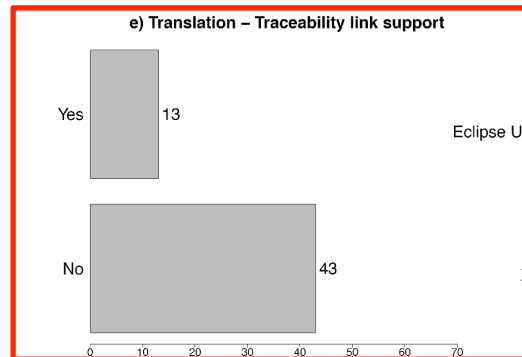
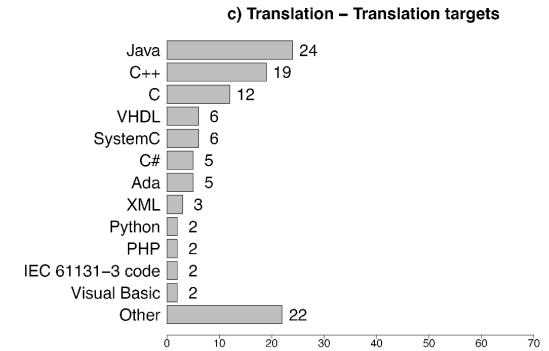
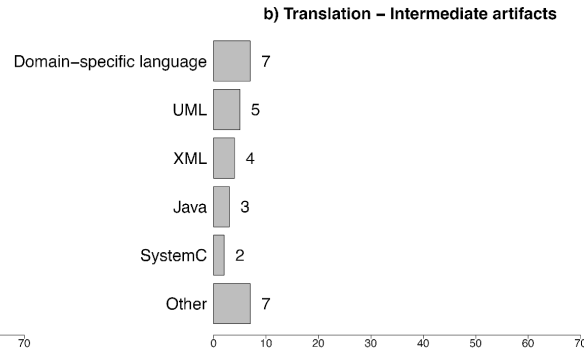
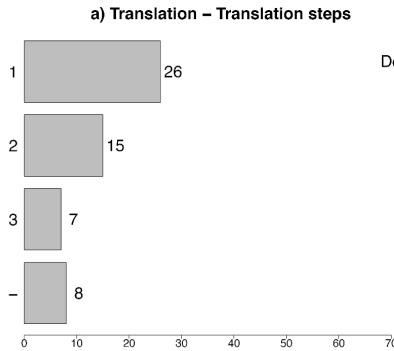
Results: Technical characteristics

- Very few solutions provide support for model-level debugging



Results: Technical characteristics

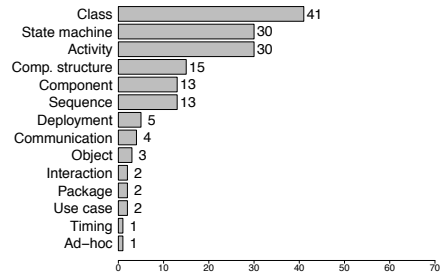
- Very few solutions explicit traceability for consistent navigation from models to code (and viceversa)



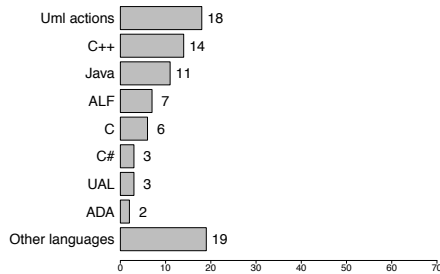
Results: Technical characteristics

- Many solutions based on ad-hoc profiles

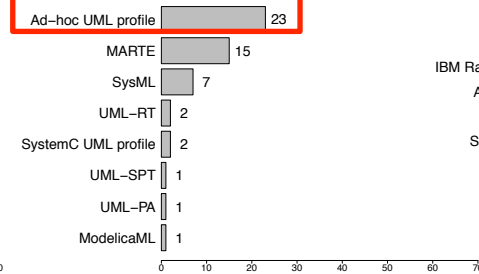
a) Required UML diagrams



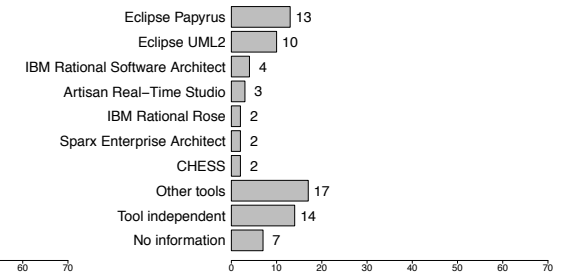
b) Used action languages



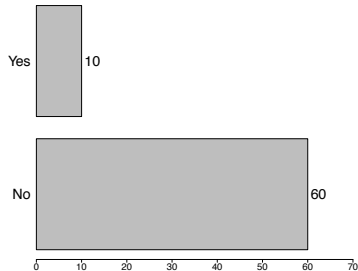
c) Explicitly required UML profiles



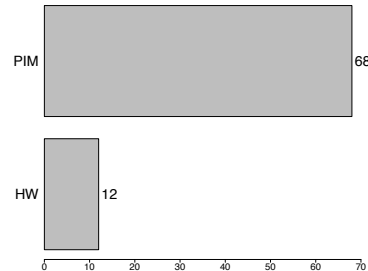
d) Modeling tool



e) Based on the fUML standard



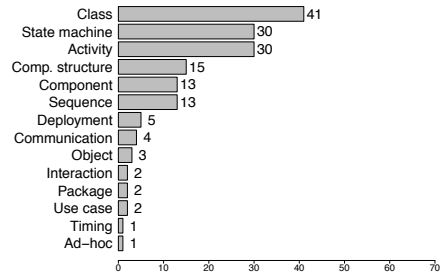
f) MDA levels covered



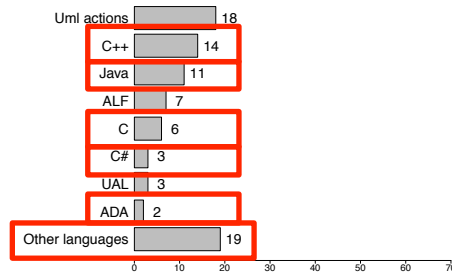
Results: Technical characteristics

- Many solutions leverage non-standard action languages

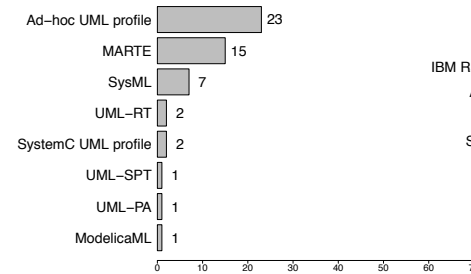
a) Required UML diagrams



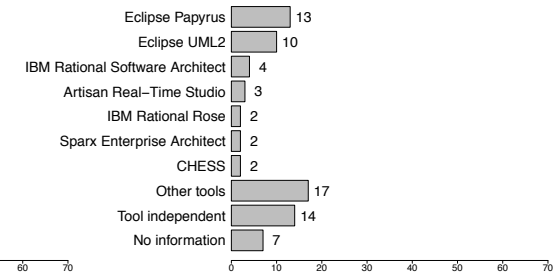
b) Used action languages



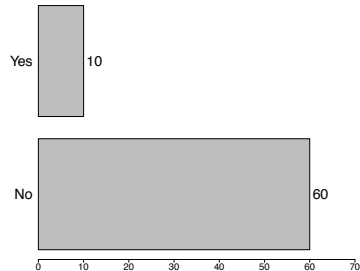
c) Explicitly required UML profiles



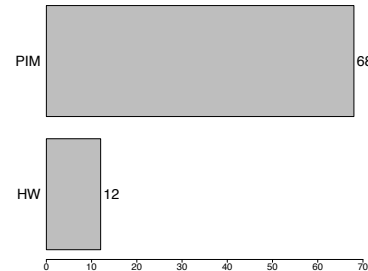
d) Modeling tool



e) Based on the fUML standard



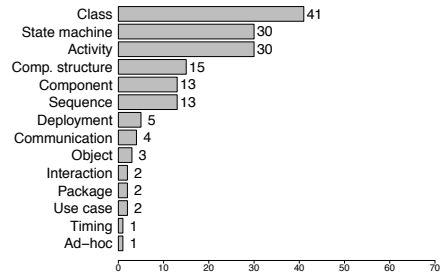
f) MDA levels covered



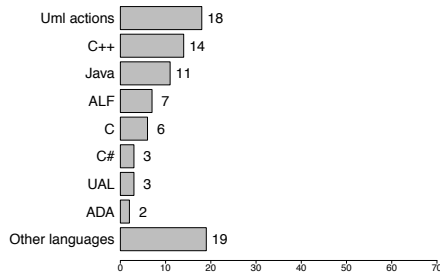
Results: Technical characteristics

- Very few solutions based on fUML (growing)

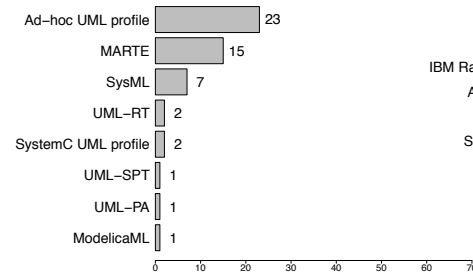
a) Required UML diagrams



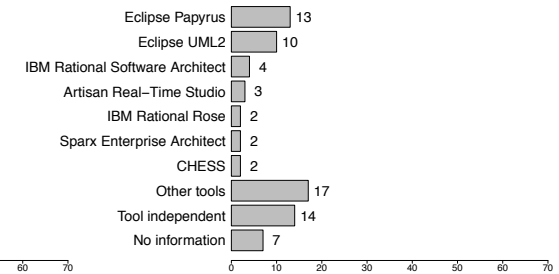
b) Used action languages



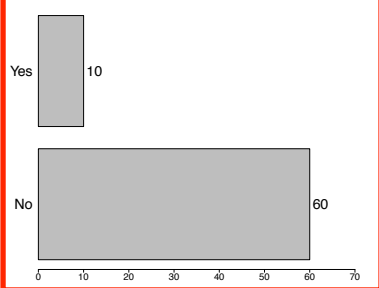
c) Explicitly required UML profiles



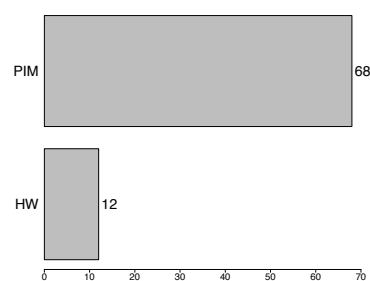
d) Modeling tool



e) Based on the fUML standard



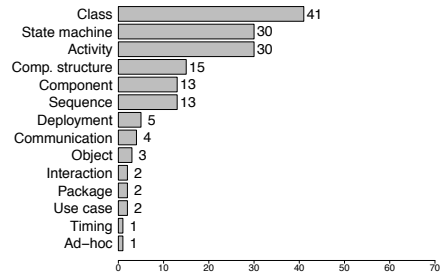
f) MDA levels covered



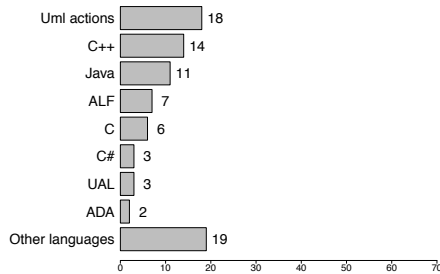
Results: Technical characteristics

- Papyrus seems to be the most used modelling tool

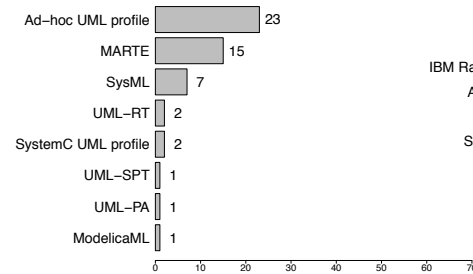
a) Required UML diagrams



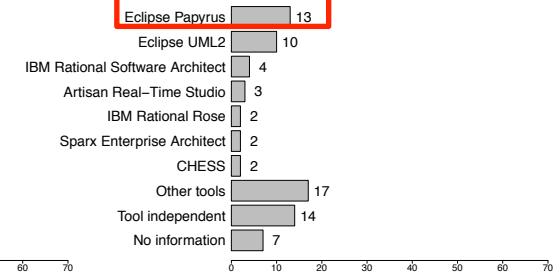
b) Used action languages



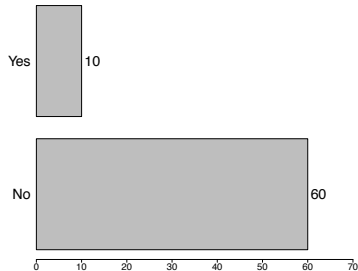
c) Explicitly required UML profiles



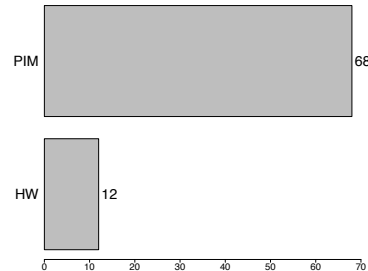
d) Modeling tool



e) Based on the fUML standard

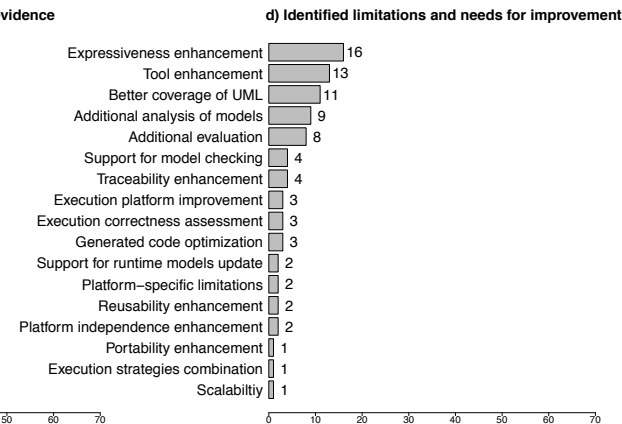
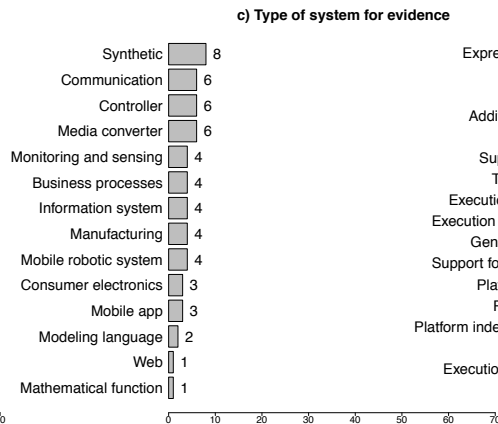
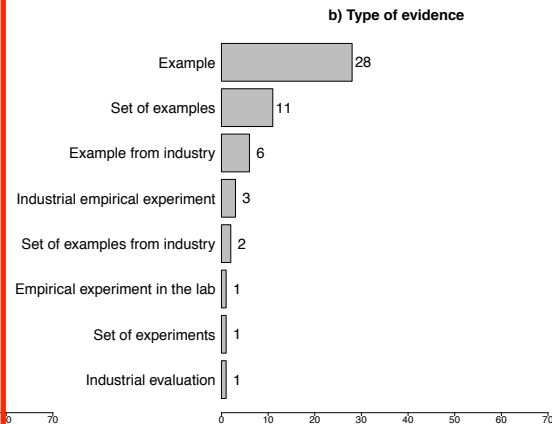
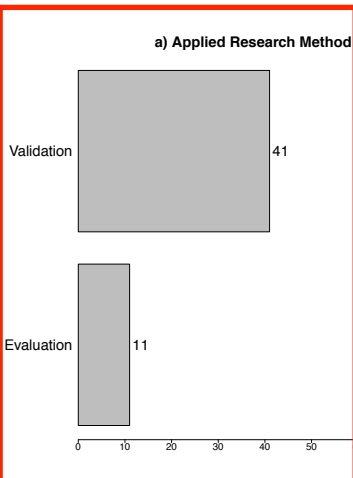


f) MDA levels covered



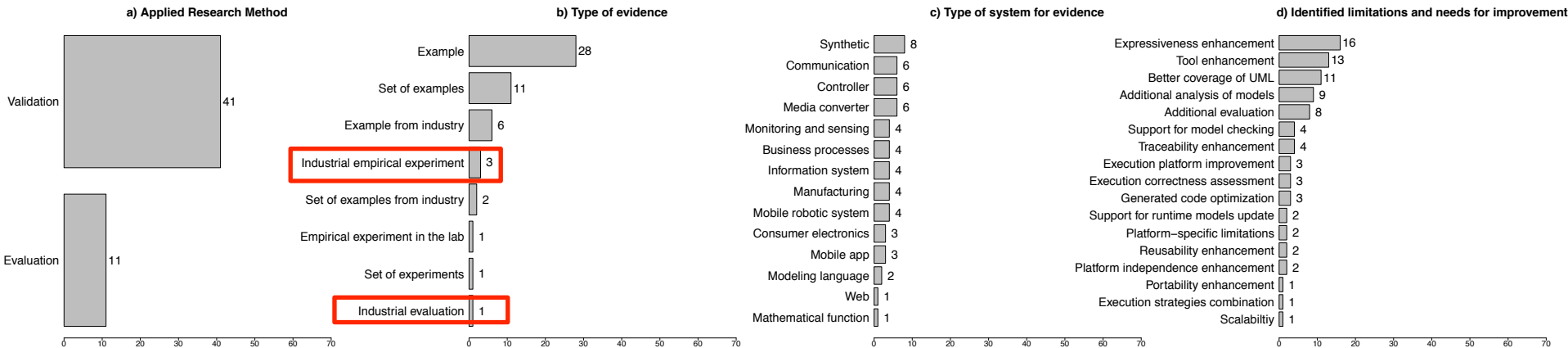
Results: Provided evidence and limitations

- The majority of the analyzed studies provide validation rather than evaluation



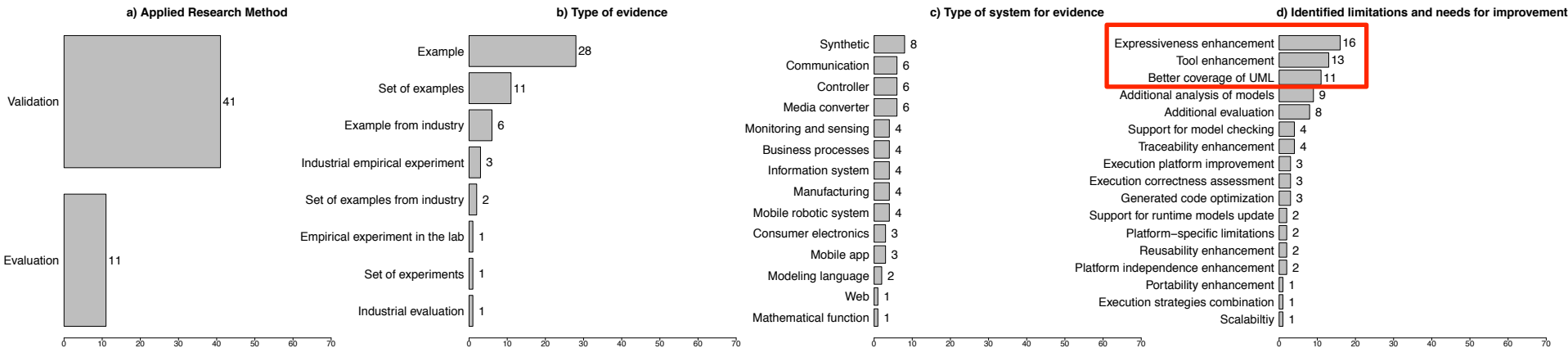
Results: Provided evidence and limitations

- A small number of studies provides evidence by experimentation in industrial settings; among them, only a few rely on empirical evaluation



Results: Provided evidence and limitations

- The most common limitations are
 - supported expressiveness in terms of covered UML concepts
 - inadequate tool support





Our Perspectives for UML execution

- The results of our study tell us that the following should be given priority both by researchers and tool developers
 - ability to execute abstract (high-level) and incomplete models



Our Perspectives for UML execution

- The results of our study tell us that the following should be given priority both by researchers and tool developers
 - ability to execute abstract (high-level) and incomplete models
 - enhanced observability of executing models



Our Perspectives for UML execution

- The results of our study tell us that the following should be given priority both by researchers and tool developers
 - ability to execute abstract (high-level) and incomplete models
 - enhanced observability of executing models
 - enhanced control of model execution



Our Perspectives for UML execution

- The results of our study tell us that the following should be given priority both by researchers and tool developers
 - ability to execute abstract (high-level) and incomplete models
 - enhanced observability of executing models
 - enhanced control of model execution
 - directly compiled model executables



Our Perspectives for UML execution

- The results of our study tell us that the following should be given priority both by researchers and tool developers
 - ability to execute abstract (high-level) and incomplete models
 - enhanced observability of executing models
 - enhanced control of model execution
 - directly compiled model executables
 - support for UML-compliant action languages



Our Perspectives for UML execution

- The results of our study tell us that the following should be given priority both by researchers and tool developers
 - ability to execute abstract (high-level) and incomplete models
 - enhanced observability of executing models
 - enhanced control of model execution
 - directly compiled model executables
 - support for UML-compliant action languages
 - support for executing models based on UML profiles



Our Perspectives for UML execution

- The results of our study tell us that the following should be given priority both by researchers and tool developers
 - ability to execute abstract (high-level) and incomplete models
 - enhanced observability of executing models
 - enhanced control of model execution
 - directly compiled model executables
 - support for UML-compliant action languages
 - support for executing models based on UML profiles
 - integration of UML simulators into heterogeneous (multi-paradigm) simulation systems



Our Perspectives for UML execution

- The results of our study tell us that the following should be given priority both by researchers and tool developers
 - ability to execute abstract (high-level) and incomplete models
 - enhanced observability of executing models
 - enhanced control of model execution
 - directly compiled model executables
 - support for UML-compliant action languages
 - support for executing models based on UML profiles
 - integration of UML simulators into heterogeneous (multi-paradigm) simulation systems
 - use of model-reference approaches at runtime



***Thanks for your
attention!***

Contact: federico.ciccozzi@mdh.se

Primary studies I

ID	Title	Author	Venue	Year
PS1	Model-Driven Design of Network Aspects of Distributed Embedded Systems	Ebeid, E. et al.	IEEE Transactions on Integrated Circuits and Systems	2015
PS2	HDL code generation from UML/MARTE sequence diagrams for verification and synthesis	Ebeid, E. et al.	Design Automation for Embedded Systems	2015
PS3	Formal verification and validation of embedded systems: the UML-based MADES approach	Baresi, L. et al.	Software & Systems Modeling	2015
PS4	Animated simulation of integrated UML behavioral models based on graph transformation	Ermel, C. et al.	IEEE Symposium on Visual Languages and Human-Centric Computing	2015
PS5	On the Generation of Full-Fledged Code from UML Profiles and ALF for Complex Systems	Ciccozzi, F. et al.	International Conference on Information Technology-New Generations	2015
PS6	Formalizing execution semantics of UML profiles with fUML models	Tatibouet, J. et al.	Model-Driven Engineering Languages and Systems	2014
PS7	A Model-Driven Approach to Generate Mobile Applications for Multiple Platforms	Usman, M. et al.	Asia-Pacific Software Engineering Conference	2014
PS8	Improving the design flow for parallel and heterogeneous architectures running real-time applications: The PHARAON FP7 project	Posadas, H. et al.	Microprocessor and Microsystems	2014
PS9	Model-driven engineering of Manufacturing Automation Software Projects A SysML-based approach	Vogel-Heuser, B. et al.	Mechatronics	2014
PS10	Extending UML/MARTE to Support Discrete Controller Synthesis, Application to Reconfigurable Systems-on-Chip Modeling	Guillet, S. et al.	ACM Transactions on Reconfigurable Technology and Systems	2014
PS11	Textual, executable, translatable UML	Dévai, G. et al.	OCL@ MoDELS	2014
PS12	Reliable execution of statechart-generated correct embedded software under soft errors	Ferreira, R.R. et al.	Design and Diagnostics of Electronic Circuits & International Symposium on Systems	2014
PS13	Combining fUML and profiles for non-functional analysis based on model execution traces	Berardinelli, L. et al.	International ACM Sigsoft conference on Quality of software architectures	2013
PS14	Executing and debugging UML models: an fUML extension	Laurent, Y. et al.	ACM Symposium on Applied Computing	2013
PS15	Environment modeling and simulation for automated testing of soft real-time embedded software	Iqbal, M.Z. et al.	Software & Systems Modeling	2013
PS16	A model-based framework for developing real-time safety Ada systems	Salazar, E. et al.	Reliable Software Technologies-Ada-Europe	2013
PS17	Multi-Paradigm Semantics for Simulating SysML Models using SystemC-AMS	Chaves Cafe, D. et al.	Specification & Forum on Design Languages	2013
PS18	An Optimized Compilation of UML State Machines	Charfi, A. et al.	International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing	2012
PS19	Symbolic execution of UML-RT state machines	Zurowska, K. et al.	ACM Symposium on Applied Computing	2012
PS20	Achieving process modeling and execution through the combination of aspect and model-driven engineering approaches	Bendraou, R. et al.	Journal of Software: Evolution and Process	2012
PS21	A Model Driven Approach for Android Applications Development	Parada, A.G. et al.	Brazilian Symposium on Computing System Engineering	2012
PS22	Modeling and simulation of secure wireless sensor network	Diaz, A. et al.	Forum on Specification and Design Languages	2012
PS23	A Plug-in Based Approach for UML Model Simulation	Radjenovic, A. et al.	European conference on Modelling Foundations and Applications	2012
PS24	On the Performance of UML State Machine Interpretation at Runtime	Höfig, E. et al.	International Symposium on Software Engineering for Adaptive and Self-Managing Systems	2011
PS25	Contracts for Model Execution Verification	Cariou, E. et al.	European conference on Modelling foundations and applications	2011
PS26	Modelica code generation from ModelicaML state machines extended by asynchronous communication	Pohlmann, U. et al.	International Workshop on Equation-Based Object-Oriented Modeling Languages and Tools	2011
PS27	Framework to Simulate the Behavior of Embedded Real-Time Systems Specified in UML Models	Wehrmeister, M.A. et al.	Brazilian Symposium on Computing System Engineering	2011
PS28	Code generation for UML 2 activity diagrams: Towards a comprehensive model-driven development approach	Gessenharter, D. et al.	European conference on Modelling foundations and applications	2011
PS29	Closing the Gap between UML-based Modeling, Simulation and Synthesis of Combined HW/SW Systems	Mischkalla, F. et al.	Design, Automation & Test in Europe	2010

Primary studies II

PS30	SystemC/C-Based Model-Driven Design for Embedded Systems	Riccobene, E. et al.	ACM Transactions on Embedded Computing Systems	2009
PS31	A co-design approach for embedded system modeling and code generation with UML and MARTE	Vidal, J. et al.	Design, Automation & Test in Europe	2009
PS32	Matilda: A Generic and Customizable Framework for Direct Model Execution in Model-Driven Software Development	Wada, H. et al.	Handbook of Research on Software Engineering and Productivity Technologies: Implications of Globalization	2009
PS33	Realization of UML class and state machine models in the C# code generation and execution framework	Derezinska, A. et al.	Informatica	2009
PS34	eUDEVS: Executable UML with DEVS Theory of Modeling and Simulation	Risco-Martin, J.L. et al.	Simulation	2009
PS35	Model-driven development of composite context-aware web applications	Kapitsaki, G.M. et al.	Information and Software Technology	2009
PS36	Performance evaluation of UML2-modeled embedded streaming applications with system-level simulation	Arpinen, T. et al.	EURASIP Journal on Embedded Systems	2009
PS37	SecureMDD: A Model-Driven Development Method for Secure Smart Card Applications	Moebius, N. et al.	International Conference on Availability, Reliability and Security	2009
PS38	Execution and Simulation of (Profiled) UML Models using Pópulo	Fuentes, L. et al.	International workshop on Models in software engineering	2008
PS39	Towards a UML virtual machine: implementing an interpreter for UML 2 actions and activities	Crane M.L. et al.	Conference of the center for advanced studies on collaborative research: meeting of minds	2008
PS40	Automatic Performance Model Transformation from UML to C++	Pllana, S. et al.	International Conference on Parallel Processing - Workshops	2008
PS41	An Execution Framework for MARTE-based Models	Mraidha, C. et al.	International Conference on Engineering of Complex Computer Systems	2008
PS42	UJECTOR: A Tool for Executable Code Generation from UML Models	Usman, M. et al.	Advanced Software Engineering and Its Applications	2008
PS43	MDD4SOA: Model-Driven Service Orchestration	Mayer, P. et al.	International IEEE Enterprise Distributed Object Computing Conference	2008
PS44	A Model-Based Approach for Platform-Independent Binary Components with Precise Timing and Fine-Grained Concurrency	Schattkowsky, T. et al.	Hawaii International Conference on System Sciences	2007
PS45	FSMC+, a tool for the generation of Java code from statecharts	Tiella, R. et al.	International Symposium on Principles and Practice of Programming in Java	2007
PS46	MDA-based approach for embedded software generation from a UML/MOF repository	Do Nascimento, F.A.M. et al.	Symposium on Integrated Circuits and Systems Design	2006
PS47	A Model-Based Approach for Executable Specifications on Reconfigurable Hardware	Schattkowsky, T. et al.	Design, Automation & Test in Europe	2005
PS48	Automatic Code Generation from a UML model to JEC 61131-3 and system configuration tools	Vogel-Heuse, B. et al.	International Conference on Control and Automation	2005
PS49	Embedded System Design Using Formal Model Refinement: An Approach Based on the Combined Use of UML and the B Language	Voros, N.S. et al.	Design Automation for Embedded Systems	2004
PS50	Deriving executable process descriptions from UML	Di Nitto, E. et al.	International Conference on Software Engineering	2002
PS51	A UML tool for an automatic generation of simulation programs	Arief, L.B. et al.	International Workshop on Software and Performance	2000
PS52	Testing and simulating production control systems using the Fujaba environment	Niere, J. et al.	Applications of Graph Transformations with Industrial Relevance	2000

Primary tools

ID	Name	URL
PS53	ARTISAN Studio Sysim	http://www.atego.com/products/sysim/
PS54	BOUML	http://www.bouml.fr/
PS55	Cameo Simulation toolkit	http://www.nomagic.com/products/magicdraw-addons/cameo-simulation-toolkit.html
PS56	CHESS	https://www.polarsys.org/projects/polarsys.chess
PS57	Fujaba	http://www.fujaba.de/
PS58	IBM Rational Rhapsody (family)	http://www.ibm.com/developerworks/downloads/r/rhapsodydeveloper/
PS59	IBM Rational Rose (family)	http://www-03.ibm.com/software/products/en/ratirosefami
PS60	IBM Rational Software Architect	http://www-03.ibm.com/software/products/en/ratisoftarch
PS61	IBM RSARTE	http://www-01.ibm.com/support/docview.wss?uid=swg27041556
PS62	IBM Rational Tau	http://www-03.ibm.com/software/products/en/ratitau
PS63	Abstract Solutions iUML	http://www.abstractsolutions.co.uk/PRODUCTS/iuml/
PS64	One Fact BridgePoint	https://xtuml.org/
PS65	fUML implementation	http://portal.modeldriven.org/content/download
PS66	Moka	https://wiki.eclipse.org/Papyrus/UserGuide/ModelExecution
PS67	Qompass	https://wiki.eclipse.org/Papyrus_Qompass
PS68	Quantum Leaps QM	http://www.state-machine.com/qm/
PS69	Sparx Enterprise Architect	http://www.sparxsystems.com/
PS70	Syntony	http://www7content.cs.fau.de/syntony/